

Compression of Multispectral Landsat Imagery Using the Embedded Zerotree Wavelet (EZW) Algorithm

Jerome M. Shapiro, Stephen A. Martucci, and Martin Czigler

The David Sarnoff Research Center

A Subsidiary of SRI International

Princeton, NJ 08543-5300.

Abstract

The Embedded Zerotree Wavelet (EZW) algorithm has proven to be an extremely efficient and flexible compression algorithm for low bit rate image coding [4]-[6]. The embedding algorithm attempts to order the bits in the bit stream in numerical importance and thus, a given code contains all lower rate encodings of the same algorithm. Thus, precise bit rate control is achievable and a target rate or distortion metric can be met exactly. Furthermore, the technique is fully image adaptive.

An algorithm for multispectral image compression which combines the spectral redundancy removal properties of the image-dependent Karhunen-Loeve Transform (KLT), with the efficiency, controllability and adaptivity of the Embedded Zerotree Wavelet algorithm is presented. Results are shown which illustrate the advantage of jointly encoding spectral components using the KLT and EZW.

1 Introduction

Multispectral image compression presents a set of new challenges in the area of image compression. In their raw form, multispectral images constitute a tremendous amount of data, and compression is essential for efficient data access, storage, and transmission of this class of imagery. Because there is also a large degree of interband correlation, there is potential for extremely high data compression without a large sacrifice in image quality, both subjectively and numerically.

In prior work described in [2], an image dependent Karhunen-Loeve Transform (KLT) was used to decorrelate a set of seven-band Landsat Thematic Mapper (TM) images prior to compression using a wavelet/subband coder. In the current work, the same image dependent KLT is used, but the compression engine that follows the KLT is replaced by a multiband implementation of the Embedded Zerotree Wavelet (EZW) algorithm. The EZW algorithm is a new compression algorithm that attempts to order the bits in the bit stream in numerical importance [4] - [6]. Because of the coarse to fine nature of the EZW algorithm, application to multiband images such as color or multispectral imagery involves simply including the additional wavelet coefficients for each band in the scanning used in EZW. This process is explained in more detail in Section 3.

2 Karhunen-Loeve Transform

There is typically a tremendous amount of interband correlation present in Landsat TM images since the sensors are co-located and the spectral weighting functions have some overlap. An effective way of exploiting this correlation is to compute the image-dependent KLT [2]. This involves performing an eigenvalue decomposition on the interband correlation matrix, and projecting the images, pixel-by-pixel, onto the orthonormal basis functions defined by the eigenvectors. The resulting principal component images each correspond to a different eigenvector. The amount of compression attainable depends on the eigenvalue spread, where a larger spread implies a higher coding gain. Once the interband correlation has been removed via the KLT, the resulting bands can be jointly encoded using the multiband EZW algorithm described in the next section.

Note that there is some overhead associated with the KLT that must be transmitted. In the results discussed below, the 7 means for each original band and the 49 elements of the eigenvector matrix are represented as 32-bit floating-point numbers for a fixed overhead of 1792 (56×32) bits. While this precision is probably unnecessary for large images, for example 512×512 , this overhead represents less than 0.007 bits per pixel. A larger drawback of the KLT approach is the computational burden in computing the KLT at the encoder. As discussed in [2], a fixed sub-optimal transformation, perhaps based on physical considerations, may be more practical at the cost of reduced coding gain. Alternatively, an intermediate compromise is to compute the KLT using data from the low frequency subbands of the wavelet transform for each original spectral component.

In addition to using the KLT for removal of spectral decomposition, Markas and Reif have also applied a histogram equalization technique to equalize the probability densities of the original bands [3]. Although this technique appears useful for visualization, the non-linearity effectively changes the gray scale units and amplifies the components with low spectral energy. As a result, joint bit allocation leads to unequal distortions distributed across the bands, causing the spectral components with the least energy to be encoded with the highest fidelity. Since EZW performs joint compression of all of the spectral components, unless the images are specifically compressed for visualization, histogram equalization would probably be inappropriate if uniform numerical distortion metrics are used.

3 Embedded Zerotree Wavelet Algorithm Description

3.1 Discrete Wavelet Transform

Each component is first transformed spatially using a discrete wavelet transform. The discrete wavelet transform used in this paper is identical to a hierarchical subband system, where the subbands are logarithmically spaced in frequency and represent an octave-band decomposition. This particular configuration has also been called a *QMF-pyramid* [1].

To begin the decomposition, the image is decomposed into four subbands by cascading horizontal and vertical two-channel critically sampled filterbanks. The filters used in the decomposition are scaled so that the squares of the filter coefficients sum to one. This normalization is important so that coefficients in all subbands can be compared to the same thresholds for the purpose of measuring numerical significance, since each coefficient is treated as a distinct, potentially important piece of data regardless of its scale. If orthogonal

wavelets are used, the resulting decomposition represents a unitary transformation. In practice, 9-tap symmetric QMF filters such as those in Adelson, *et. al.* [1] have been found to be effective. Note that for these QMF filters, the low-pass and high-pass filters in the filterbank are orthogonal, but these filters are only nearly orthogonal to their even-integer translates. However, for coding purposes, the discrete wavelet transform generated from these filters can be treated as unitary since the deviation from unitary is negligible compared to the quantization error.

After the first scale of the decomposition, to tile the entire image in each subband, each coefficient represents a spatial area corresponding to approximately a 2×2 area of the original picture. To tile the 2-D frequency domain, the low frequencies represent a bandwidth in each dimension approximately corresponding to $0 < |\omega| < \frac{\pi}{2}$, whereas the high frequencies represent the band from $\frac{\pi}{2} < |\omega| < \pi$. To obtain the next coarser scale of wavelet coefficients, the lowest frequency subband is further decomposed and critically sampled. The process continues until some final scale is reached. Note that at each scale, there are 3 subbands. The remaining lowest frequency subband is a representation of the information at all coarser scales. Note also that for each coarser scale, the coefficients represent a larger spatial area of the image but a narrower band of frequencies.

3.2 Successive-Approximation

To perform the embedded coding, successive-approximation quantization (SAQ) is applied. As will be seen, SAQ is related to bit-plane encoding of the magnitudes. Given an amplitude threshold T , a wavelet coefficient x is said to be *insignificant* with respect to T if $|x| < T$. The SAQ sequentially applies a sequence of thresholds T_0, \dots, T_{N-1} to determine significance, where the thresholds are chosen so that $T_i = T_{i-1}/2$. The initial threshold T_0 is chosen so that $|x_j| < 2T_0$ for all transform coefficients x_j .

During the encoding (and decoding), two separate lists of coordinates of wavelet coefficients are maintained. At any point in the process, the *dominant* list contains the *coordinates* of those coefficients that have not yet been found to be significant in the same relative order as the initial scan. This scan is such that the subbands are ordered, and within each subband, the set of coefficients are ordered. The *subordinate* list contains the *magnitudes* of those coefficients that have been found to be significant. For each threshold, each list is scanned once.

3.3 The Dominant Pass: Zerotree Coding of Significance Maps

During a *dominant pass*, coefficients with coordinates on the dominant list, i.e. those that have not yet been found to be significant, are compared to the threshold T_i to determine their significance, and if significant, their sign is also recorded. A map indicating the result of a binary (significant or insignificant) or a ternary (positive significant, negative significant or insignificant) decision is called a *significance map*. This significance map for the dominant pass is encoded using zerotree coding as outlined below.

A parent-child relationship can be defined between wavelet coefficients at different scales corresponding to the same location. With the exception of the highest frequency subbands, every coefficient at a given scale can be related to a set of coefficients at the next finer

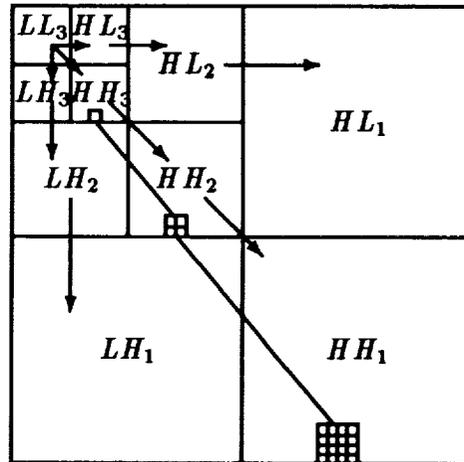


Figure 1: Parent-Child Dependencies of Subbands. Note that the arrow points from the subband of the parents to the subband of the children. The lowest frequency subband is the top left, and the highest frequency subband is at the bottom right. Also shown is a wavelet tree consisting of all of the descendants of a single coefficient in subband $HH3$. The coefficient in $HH3$ is a zerotree root if it is insignificant and *all* of its descendants are insignificant.

scale of similar orientation. The coefficient at the coarse scale will be called the *parent*, and all coefficients corresponding to the same spatial location at the next finer scale of similar orientation will be called *children*. The parent-child dependencies are shown in Fig. 1. With the exception of the lowest frequency subband, all parents have four children. For the lowest frequency subband, the parent-child relationship is defined such that each parent has three children, one in each subband at the same scale.

The scanning of the coefficients processed during a dominant pass is performed in such a way that no child is scanned before its parent. For an N -scale pyramid, the scan begins at the lowest frequency subband, denoted as LL_N , and scans subbands LH_N , HL_N , and HH_N , at which point it moves on to scale $N - 1$, etc. Note that each coefficient within a given subband is considered before the scan moves to the next subband.

Given a threshold level T_i to determine whether or not a coefficient is significant, a coefficient x is said to be an element of a *zerotree* if it is insignificant and all of its descendants are also insignificant. A coefficient is said to be a *zerotree root* for a threshold T_i if 1) the coefficient is insignificant, 2) the coefficient is not the descendant of a previously found zerotree root for T_i , i.e. it is not *predictably insignificant* from the discovery of a zerotree root at a coarser scale, and 3) *all* of its descendants are insignificant.

During the scanning of the coefficients during a dominant pass, each coefficient that is not predictably insignificant is encoded with a symbol from the four symbol alphabet: 1) zerotree root, 2) isolated zero, 3) positive significant, and 4) negative significant, where an isolated zero implies that the coefficient under consideration is insignificant but has a significant descendant. The string of symbols is then encoded using a multi-level adaptive arithmetic coder such as in Witten, *et. al* [7]. Each time a coefficient is encoded as significant,

(positive or negative), its magnitude is appended to the subordinate list. Also note that once a coefficient is determined to be significant, for the purpose of determining if one of its ancestors is a zerotree on future dominant passes, its value is treated as zero so as not to prevent a zerotree occurrence on future dominant passes.

3.4 The Subordinate Pass: Refinement of Significant Coefficients

A dominant pass is followed by a *subordinate pass* in which all coefficients on the subordinate list are scanned and the specifications of the magnitudes available to the decoder are refined to an additional bit of precision. More specifically, during a subordinate pass, the *width* of the effective quantizer step size, which defines an uncertainty interval for the true magnitude of the coefficient, is cut in half. For each magnitude on the subordinate list, this refinement can be encoded using a binary alphabet with a “1” symbol indicating that the true value falls in the upper half of the old uncertainty interval and a “0” symbol indicating the lower half. The string of symbols from this binary alphabet that is generated during a subordinate pass is then entropy coded. Note that prior to this refinement, the width of the uncertainty region is exactly equal to the current threshold. After the completion of a subordinate pass the magnitudes on the subordinate list are sorted in decreasing magnitude, to the extent that the decoder has the information to perform the same sort.

3.5 Embedded Coding

The process continues to alternate between dominant passes and subordinate passes where the threshold is halved before each dominant pass. (In principle one could divide by other factors than 2. This factor of 2 was chosen here because it has nice interpretations in terms of bit plane encoding and numerical precision in a familiar base 2, and good coding results were obtained).

In the decoding operation, each decoded symbol, both during a dominant and a subordinate pass, refines and reduces the width of the uncertainty interval in which the true value of the coefficient (or coefficients, in the case of a zerotree root) may occur. The reconstruction value used can be anywhere in that uncertainty interval. For minimum mean-square error distortion, one could use the centroid of the uncertainty region using some model for the PDF of the coefficients. However, a practical approach is to simply use the center of the uncertainty interval as the reconstruction value.

The encoding stops when some target stopping condition is met, such as when the bit budget is exhausted. The encoding can cease at any time and the resulting bit stream contains all lower rate encodings. Note that if the bit stream is truncated at an arbitrary point, there may be bits at the end of the code that do not decode to a valid symbol since a codeword has been truncated. In that case, these bits do not reduce the width of an uncertainty interval or any distortion function. In fact, it is very likely that the first L bits of the bit stream will produce exactly the same image as the first $L + 1$ bits which occurs if the additional bit is insufficient to complete the decoding of another symbol. Nevertheless, terminating the decoding of an embedded bit stream at a specific point in the bit stream produces exactly the same image would have resulted had that point been the initial target rate. This ability to cease encoding or decoding anywhere is extremely useful in systems

that are either rate-constrained or distortion-constrained. A side benefit of the technique is that an operational rate vs. distortion plot for the algorithm can be computed on-line.

Compression is achieved both by eliminating a large number of predictably insignificant coefficients from consideration through zerotree coding, and by adaptively arithmetic coding a string of symbols from a small alphabet. Note that the small size of the alphabet poses a tremendous advantage for an adaptive coder. Since all possible events usually occur with easily measurable frequency, an adaptation algorithm with a short memory can learn quickly and constantly track changing symbol probabilities. This adaptivity accounts for some of the effectiveness of the overall algorithm. Contrast this with the case of a large alphabet, as is the case in algorithms that don't use successive approximation. In that case, it takes many events before an extremely unlikely symbol occurs, and there are usually very many unlikely symbols. Furthermore, the probability estimates for rare events in a large alphabet are fairly unreliable because images are typically statistically non-stationary and local symbol probabilities change from region to region. Thus, the advantage of a small alphabet in an adaptive coder is that no coding capacity is wasted accounting for the possible occurrence of a large number of rare events.

3.6 Multiband EZW

Extension of the EZW algorithm to handle multispectral imagery is accomplished by simply including the wavelet transform of each principal component in the scan of the dominant pass. The scanning begins on the lowest frequency subband of the wavelet transform of the principal component corresponding to the largest eigenvalue. This entire component is scanned at a given threshold after which the scanning continues for each component in order of decreasing eigenvalue. Thus, a dominant pass for a given threshold involves scanning the transforms of all of the components at the same significance level. Although each component is scanned independently during a dominant pass, the magnitudes of significant coefficients are all placed on the *same* subordinate list. As a consequence, the refinement of significant coefficients on a subordinate pass makes no distinction as to which component a coefficient originated from. Although statistically the components corresponding to small eigenvalues contain little energy, if there are wavelet coefficients of these components that are large, bits will automatically be allocated to correctly represent their significance.

4 Experimental Results

The same Landsat 5 TM images of Kuwait that were used in [2] were again used in this new study. In addition, experiments were run using the Landsat images of Washington, D.C. All images were obtained from the USGS EROS Data Center (Sioux Falls, SD). As explained in [2], the Landsat TM data was produced by 7 sensors, where each sensor generates one band of imagery data. Bands 1 to 3 correspond to visible spectra, Band 4 to near IR spectra, Bands 5 and 7 to mid IR spectra, and Band 6 to thermal spectra. The instantaneous field of view (IFOV) for all sensors is about 30×30 m, except for Band 6, which has an IFOV of 120×120 m. All images are of size 512×512 pixels at 8 bits/pixel.

The sequence of steps for this new method of compressing multispectral data that were followed in this study are:

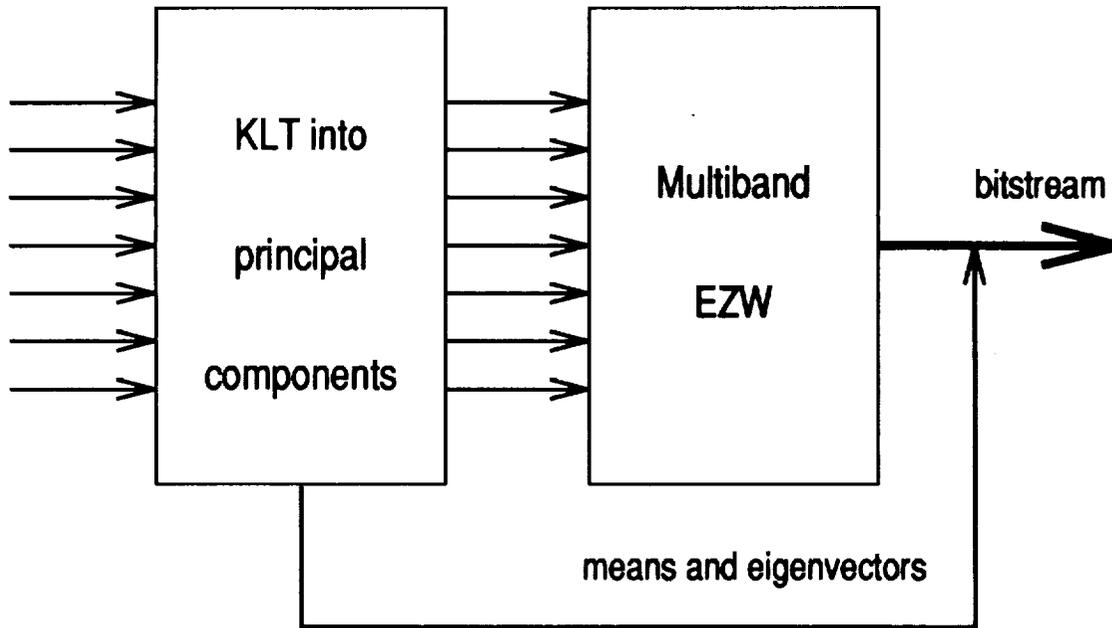


Figure 2: Block Diagram of KLT-EZW Encoder

1. Calculate then subtract the mean from each spectral band.
2. Calculate then apply KLT across all spectral bands to transform into principal components.
3. Compress principal components to target bit rate using the multispectral EZW algorithm.
4. Transmit means and eigenvectors as overhead.
5. Decompress bitstream using the multispectral EZW algorithm to recover the principal components.
6. Apply inverse KLT to transform principal components back into spectral bands.
7. Add mean to each band; reconstructed spectral bands result.

A block diagram of the encoder portion of the multispectral compression system is given in Fig. 2.

To evaluate the effectiveness of the new compression scheme, the mean square error between each original spectral band image and its reconstruction was calculated. These errors were then summed over all 7 bands. The totals are given in Table 1 for the Kuwait data under the heading *Principal Components* and subheading *new method* and in Table 2 for the Washington data under the heading *Principal Components*. The results reported in [2] are also included in Table 1 under the subheading *old method*. The bit rates shown in the table are the same as those reported in [2]. In that earlier study, the degree of compression was controlled by the specification of the quantizer bin sizes. Rate control was not used, and the bit rate of the encoded bitstream was just a consequence of the bin sizes. In the new

Table 1: Mean Square Error Results for Compression of Kuwait Images.

<i>bits/pixel</i>	<i>Original Bands</i>		<i>Principal Components</i>	
	<i>old method</i>	<i>new method</i>	<i>old method</i>	<i>new method</i>
2.51	40.02	31.63	N/A	14.13
1.55	N/A	52.04	25.11	20.24
1.06	73.82	62.45	N/A	29.16
0.73	N/A	83.71	47.96	38.28

Table 2: Mean Square Error Results for Compression of Washington Images.

<i>bits/pixel</i>	<i>Original Bands</i>	<i>Principal Components</i>
4.0	42.72	28.51
2.0	81.18	51.92
1.0	113.89	77.38
0.5	152.54	113.94

method, any desired bit rate can be met exactly; there is no need for explicit rate control. Thus, the mean square error results of the new method can be compared directly to those of the old method because the compression could be done to the same bit rates.

Experiments were also done to assess the performance of the multispectral EZW algorithm without first computing the principal components. The mean square errors of the resulting compressed images are given in the tables under the heading *Original Bands*.

As can be seen in the table, the new method gives significantly better performance than the old method, both when the principal components are not used and when they are. Even more significant is the improvement obtained by making use of the principal components. Thus, there are gains due to the multispectral EZW algorithm itself as well as gains due to transforming the imagery into its principal components.

5 Conclusion

Spectral decorrelation using an image dependent KLT followed by compression using the multiband EZW algorithm is an effective way to jointly encode the spectral bands of multispectral images. In contrast to the independent coding of the principal component images that was used in [2], the EZW algorithm jointly optimizes the bit allocation uniformly across all of the bands. Furthermore, the embedding and adaptivity features inherent in EZW allow precise rate control and eliminate the need to train the coder for a particular class of imagery.

References

- [1] E. H. Adelson, E. Simoncelli, and R. Hingorani, "Orthogonal pyramid transforms for image coding", *Proc. SPIE*, vol. 845, Cambridge, MA, pp. 50-58, Oct. 1987.
- [2] B. R. Epstein, R. Hingorani, J. M. Shapiro, and M. Czigler, "Multispectral KLT-Wavelet Data Compression for Landsat Thematic Mapper Images", *Proc. Data Compression Conf.*, Snowbird, Utah, IEEE Computer Society Press, 1992.
- [3] T. Markas and J. Reif, "Multispectral Image Compression Algorithm", *Proc. Data Compression Conf.*, Snowbird, Utah, IEEE Computer Society Press, 1993.
- [4] J. M. Shapiro, "An Embedded Wavelet Hierarchical Image Coder", *Proc. IEEE Int. Conf. Acoust., Speech, Signal Proc.*, San Francisco, CA, March 1992.
- [5] J. M. Shapiro, "An embedded hierarchical image coder using zerotrees of wavelet coefficients", *Proc. Data Compression Conf.*, Snowbird, Utah, IEEE Computer Society Press, 1993.
- [6] J. M. Shapiro, "Embedded Image Coding using Zerotrees of Wavelet Coefficients", to appear *IEEE Trans. Signal Processing, Special Issue on Wavelets and Signal Processing*, scheduled for Dec. 1993.
- [7] I. H. Witten, R. Neal, and J. G. Cleary, "Arithmetic coding for data compression", *Comm. ACM*, vol. 30, pp. 520-540, June 1987.

